ʕ •ᴥ• ʔ

b34r5hell

Steganography and Forensics

Agenda

```
> What is Steganography?
> steghide and other tools
> What is Forensics?
> Wireshark and other tools
> CTF Examples
```

ʕ•ᴥ•ʔ

Follow Along

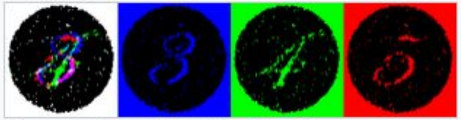
- Similar to last time, we will be going over the exercises under steg-forensics in the Bootcamp Github
- Need the Docker setup and Wireshark
- Made some small changes to the Docker setup so run “git pull”

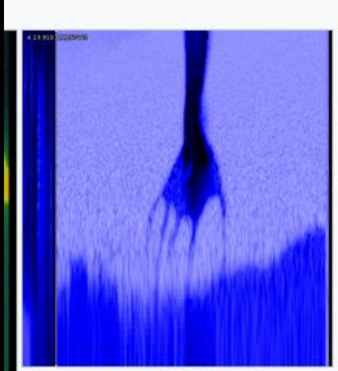
What is Steganography?

Hiding Information in Unexpected Places

- **Physical Examples**
  - Invisible Ink
  - Morse-code via body movements

- **Digital Examples (What we care about)**
  - Using certain bits of each pixel to hide a file within an image
  - Text encoded in audio (spectrogram)
  - Used in combination with cryptography


The same image viewed by white, blue, green, and red lights reveals different hidden numbers.


Spectrogram of a hidden image encoded as sound in the song "My Violent Heart" by Nine Inch Nails from the *Year Zero* album (2007)

ʕ•ᴥ•ʔ

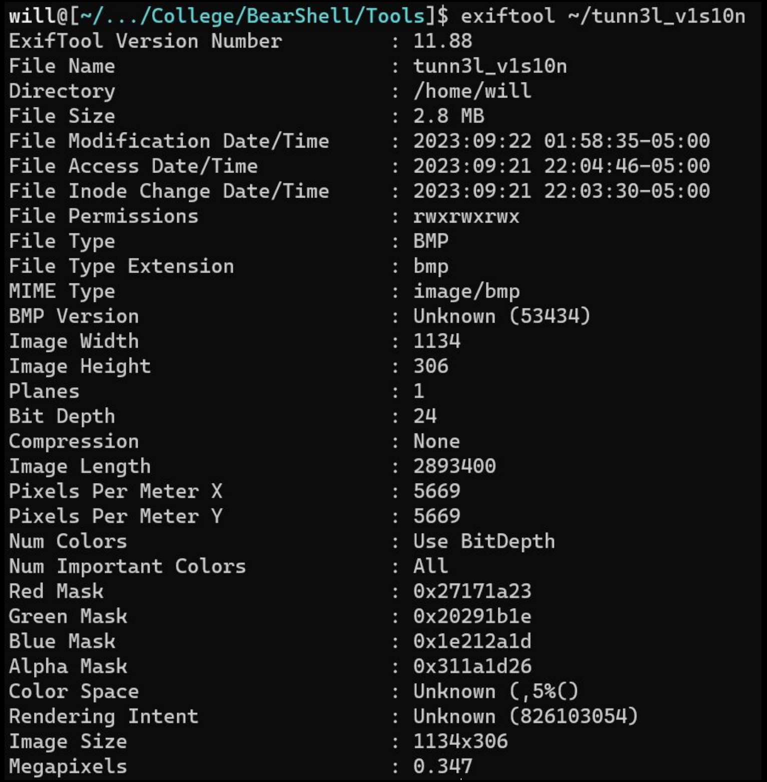
Basic Steganography Techniques

- **strings** - command line tool
  - Treats any file as text-based and finds human-readable strings
  - Will detect string-data embedded inside a file
  - When the flag format is known, you can search for it in the output
  - Can provide useful information even if it’s not the flag

- What if we break up the data?
- Put each byte of the data linearly in the image by setting the LSB or MSB of each pixel color
  - LSB is better than MSB, why?
- Archives can be directly embedded within files. Use **binwalk** to test this
  - Why will strings not work on this?
  - Also look for file and directory names hidden in files, as zip archives store these uncompressed

exiftool

- Command line tool that extracts metadata from files
- Useful for finding hidden files or providing a guide on the next step for the steg problem



```
will@[~/.../College/BearShell/Tools]$ exiftool ~/tunn3l_v1s10n
ExifTool Version Number         : 11.88
File Name                       : tunn3l_v1s10n
Directory                       : /home/will
File Size                       : 2.8 MB
File Modification Date/Time     : 2023:09:22 01:58:35-05:00
File Access Date/Time           : 2023:09:21 22:04:46-05:00
File Inode Change Date/Time     : 2023:09:21 22:03:30-05:00
File Permissions                : rwxrwxrwx
File Type                       : BMP
File Type Extension             : bmp
MIME Type                       : image/bmp
BMP Version                     : Unknown (53434)
Image Width                     : 1134
Image Height                    : 306
Planes                          : 1
Bit Depth                       : 24
Compression                     : None
Image Length                    : 2893400
Pixels Per Meter X              : 5669
Pixels Per Meter Y              : 5669
Num Colors                      : Use BitDepth
Num Important Colors            : All
Red Mask                        : 0x27171a23
Green Mask                      : 0x20291b1e
Blue Mask                       : 0x1e212a1d
Alpha Mask                      : 0x311a1d26
Color Space                     : Unknown (,5%()
Rendering Intent                : Unknown (826103054)
Image Size                      : 1134x306
Megapixels                      : 0.347
```

steghide

- Command line tool used to hide and extract data from image and audio files (cannot use PNG format)
    - Note: for audio files, a tool like **audacity** is useful and can generate a spectrogram of the file
- When hiding data, a passphrase can be provided to encrypt data and pseudo-randomly place the data in the file
    - Need passphrase to extract data and know if data is even hidden
    - a tool like **stegseek** can brute force passwords
- See “man steghide”

Steg Summary

- Steg is an expansive topic with many techniques and tools
    - There is no one-size fits all tool or approach
    - There is lots to learn and explore in this topic. We are only covering the basics
- For even more steg tools, see the bear-ctf resources Github (often useful when stuck during a competition)

What is Forensics?

In General: Collecting and examining evidence

In Cybersecurity: Collecting and examining digital data/activity

- **Examples**
  - Examining network traffic for patterns/irregularities
  - Analyzing suspicious files or databases
  - Recovering a corrupted file
  - Verifying authenticity of an image (“Is it edited?”)

Forensics and CTFs

- Broad category, usually more “puzzle-like”
- Overlap with things like Steganography and Cryptography
- Common Challenge Types
  - Network Packet Capture Analysis (Wireshark)
  - Memory dump analysis
  - File Format Analysis (e.g. header data)
  - Files inside of files inside of files inside of files inside of files can you even read this?

source: https://trailofbits.github.io/ctf/forensics/

Common Techniques and Tricks

- Network Traffic
    - Data can be hidden inside of unused packet header elements or disguised as a different form of traffic
        - Examples
            - DNS tunnelling (Wireshark twoo twooo two twoo)
                - In the real world, this can be used to bypass firewalls
            - Hide data in port numbers (shark on wire 2)
- Given the file with missing or incomplete file metadata
    - Examples
        - Given an array of pixels, must process them into an image
        - File header is corrupted, must be fixed - **hexedit** is useful
    - Look for magic numbers: Used to indicate the start of a file or new data type
        - **binwalk** is designed to look for magic numbers

Wireshark

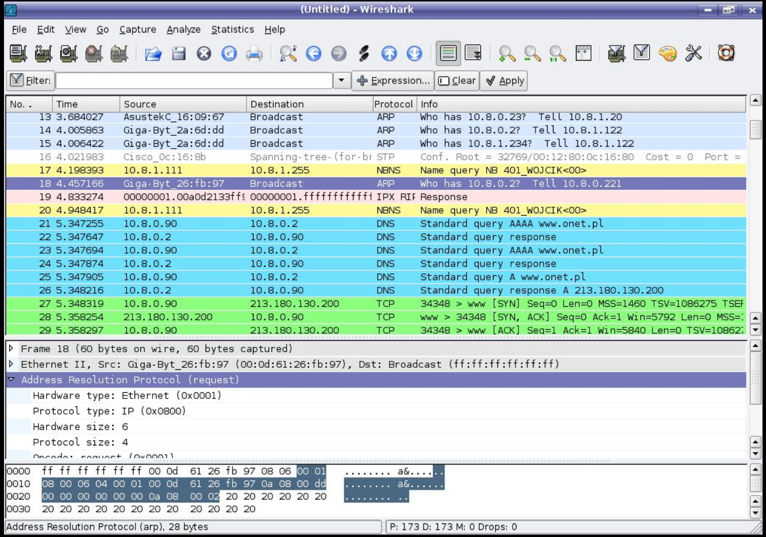
> Free/open-source network packet analyzer

> Uses PCAP (packet capture) files

> Contain a set of data packets traveling to/from devices on a network

> Various Protocols
> > HTTP (Typical Web Requests)
> > DNS (Domain Name System)
> > FTP (File Transfer Protocol)
> > *and many more*

Note: Don’t worry if you don’t know how a protocol works. Best way to learn is by doing (in other words, doing CTFs)



No.	Time	Source	Destination	Protocol	Info
13	3.684027	AsustekC_16:09:67	Broadcast	ARP	Who has 10.8.0.23? Tell 10.8.1.20
14	4.005863	Giga-Byt_2a:6d:dd	Broadcast	ARP	Who has 10.8.0.2? Tell 10.8.1.122
15	4.006422	Giga-Byt_2a:6d:dd	Broadcast	ARP	Who has 10.8.1.234? Tell 10.8.1.122
16	4.021983	Cisco_0c:16:8b	Spanning-tree-(for-br	STP	Conf. Root = 32769/00:12:80:0c:16:80 Cost = 0 Port =
17	4.198393	10.8.1.111	10.8.1.255	NBNS	Name query NB 401_WOJCIK<00>
18	4.457166	Giga-Byt_26:fb:97	Broadcast	ARP	Who has 10.8.0.2? Tell 10.8.0.221
19	4.833274	00000001.00a0d2133ff	00000001.ffffffffffff	IPX RIF	Response
20	4.948417	10.8.1.111	10.8.1.255	NBNS	Name query NB 401_WOJCIK<00>
21	5.347255	10.8.0.90	10.8.0.2	DNS	Standard query AAAA www.onet.pl
22	5.347647	10.8.0.2	10.8.0.90	DNS	Standard query response
23	5.347694	10.8.0.90	10.8.0.2	DNS	Standard query AAAA www.onet.pl
24	5.347874	10.8.0.2	10.8.0.90	DNS	Standard query response
25	5.347905	10.8.0.90	10.8.0.2	DNS	Standard query A www.onet.pl
26	5.348216	10.8.0.2	10.8.0.90	DNS	Standard query response A 213.180.130.200
27	5.348319	10.8.0.90	213.180.130.200	TCP	34348 > www [SYN] Seq=0 Len=0 MSS=1460 TSV=1086275 TSEF
28	5.358254	213.180.130.200	10.8.0.90	TCP	www > 34348 [SYN, ACK] Seq=0 Ack=1 Win=5792 Len=0 MSS=:
29	5.358297	10.8.0.90	213.180.130.200	TCP	34348 > www [ACK] Seq=1 Ack=1 Win=5840 Len=0 TSV=10862;

```
▷ Frame 18 (60 bytes on wire, 60 bytes captured)
▷ Ethernet II, Src: Giga-Byt_26:fb:97 (00:0d:61:26:fb:97), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
▽ Address Resolution Protocol (request)
    Hardware type: Ethernet (0x0001)
    Protocol type: IP (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: request (0x0001)
```

```
0000  ff ff ff ff ff ff 00 0d  61 26 fb 97 08 06 00 01   ........ a&......
0010  08 00 06 04 00 01 00 0d  61 26 fb 97 0a 08 00 dd   ........ a&......
0020  00 00 00 00 00 00 0a 08  00 02 20 20 20 20 20 20   ........ ..
0030  20 20 20 20 20 20 20 20  20 20 20 20
```

Address Resolution Protocol (arp), 28 bytes

P: 173 D: 173 M: 0 Drops: 0

tcpdump

- Command line packet capturing and PCAP analyzer/visualizer
- Similar to Wireshark just less robust and no GUI
  - Good to know that it exists, as Wireshark is not always available

Summary

- Forensics is one of the most open-ended category types
- Challenges often involve exploring a file given to you, looking for clues and the flag

- Once data has been encrypted or hidden via steganography and/or encryption, it is transferred over a network
- PCAP files represent a recording of the network traffic
- Identify what is happening in the network and how data may have been hidden to get the flag

Tasks

- Install steghide, binutils (strings), and binwalk on the Linux environment
- Complete the associated dojo module